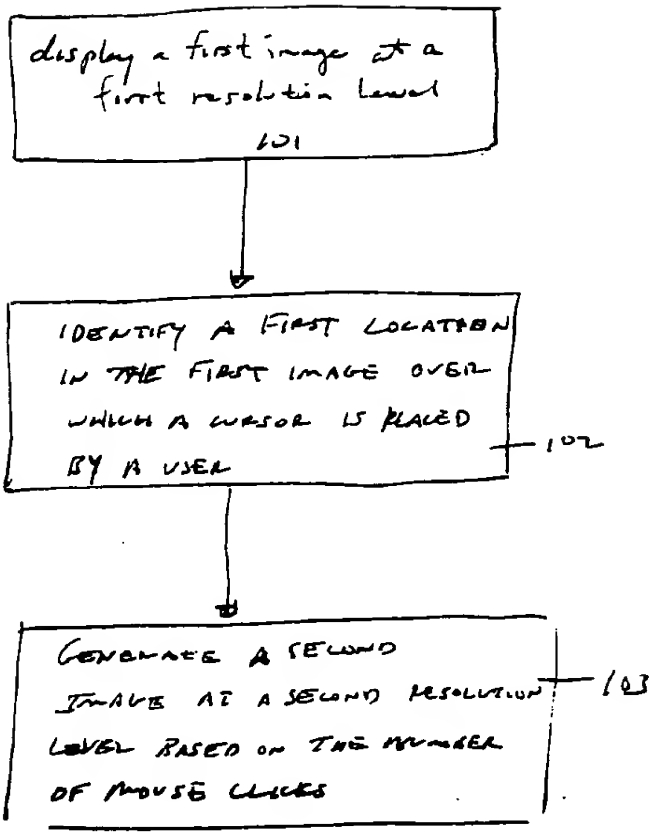

display a first image at a first resolution level
101
IDENTIFY A FIRST LOCATION IN THE FIRST IMAGE OVER WHICH A CURSOR IS PLACED BY A USER
102
GENERATE A SECOND IMAGE AT A SECOND RESOLUTION LEVEL BASED ON THE NUMBER OF MOUSE CLICKS
103

Figure 1

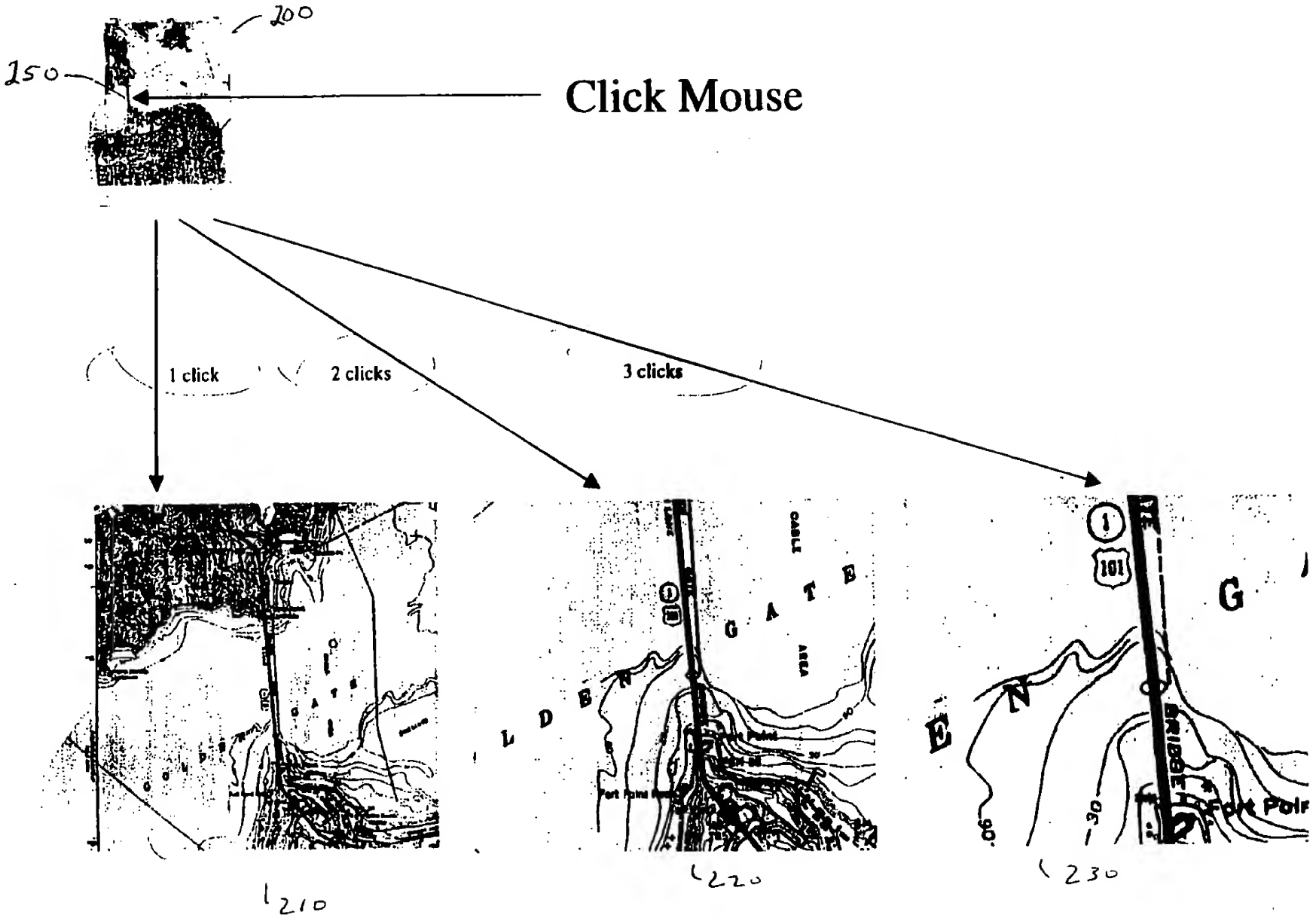


Figure 2

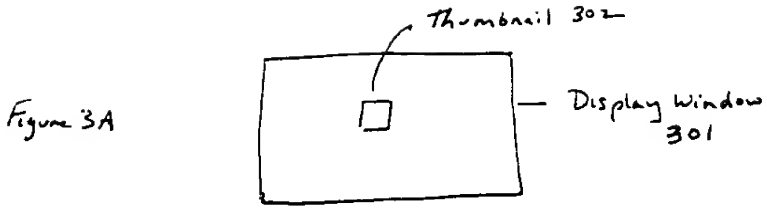


Figure 3A

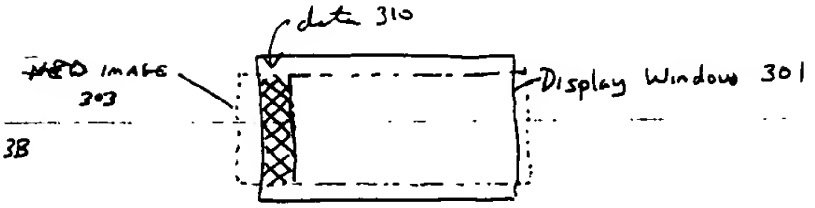


Figure 3B

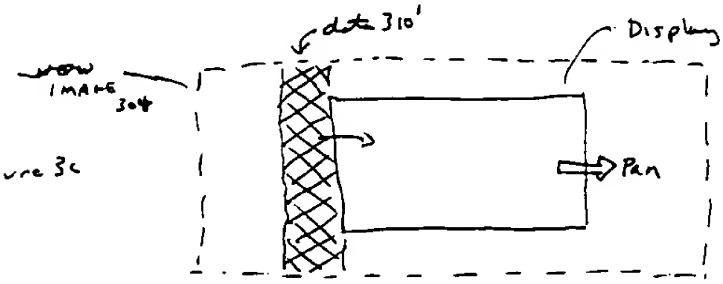


Figure 3C

Figure 3

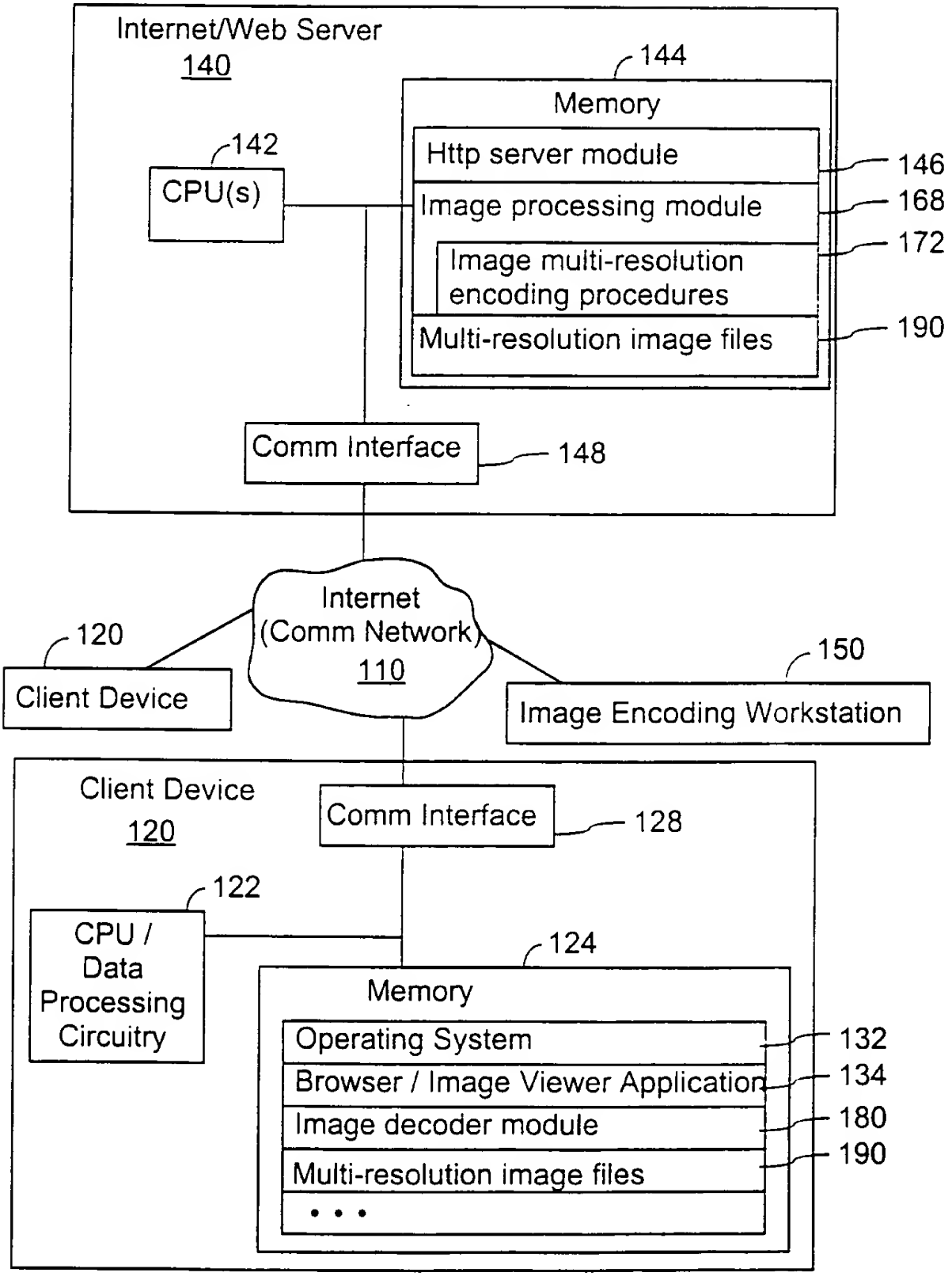

100
Internet/Web Server
140
144
Memory
142
CPU(s)
Http server module
146
Image processing module
168
Image multi-resolution encoding procedures
172
Multi-resolution image files
190
Comm Interface
148
Internet (Comm Network)
110
120
Client Device
150
Image Encoding Workstation
Client Device
120
Comm Interface
128
122
CPU / Data Processing Circuitry
124
Memory
Operating System
132
Browser / Image Viewer Application
134
Image decoder module
180
Multi-resolution image files
190
• • •

FIG. 4

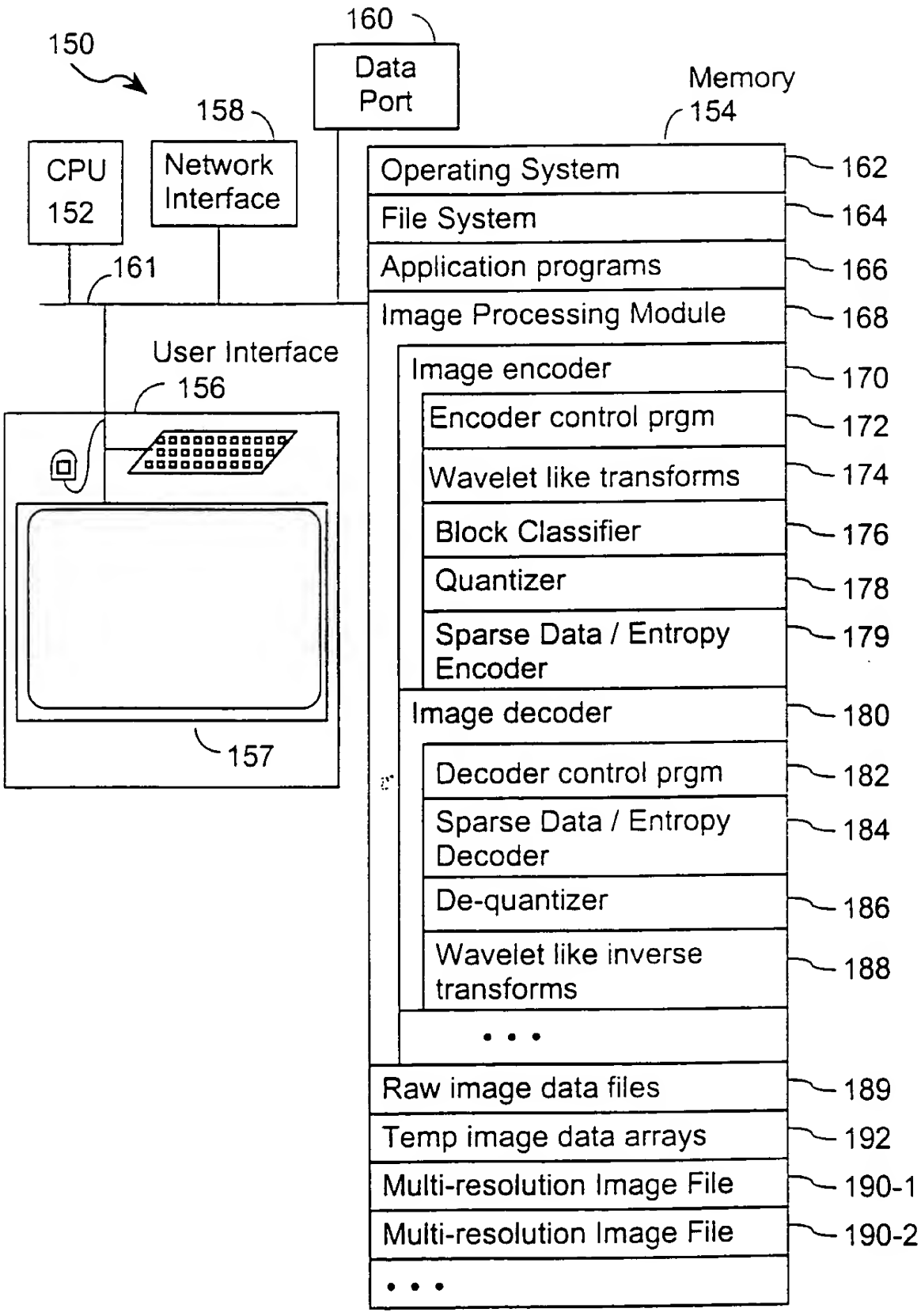

150
160
Data Port
Memory
154
158
Network Interface
CPU 152
161
User Interface
156
157
Operating System
162
File System
164
Application programs
166
Image Processing Module
168
Image encoder
170
Encoder control prgm
172
Wavelet like transforms
174
Block Classifier
176
Quantizer
178
Sparse Data / Entropy Encoder
179
Image decoder
180
Decoder control prgm
182
Sparse Data / Entropy Decoder
184
De-quantizer
186
Wavelet like inverse transforms
188
• • •
Raw image data files
189
Temp image data arrays
192
Multi-resolution Image File
190-1
Multi-resolution Image File
190-2
• • •

FIG. 5

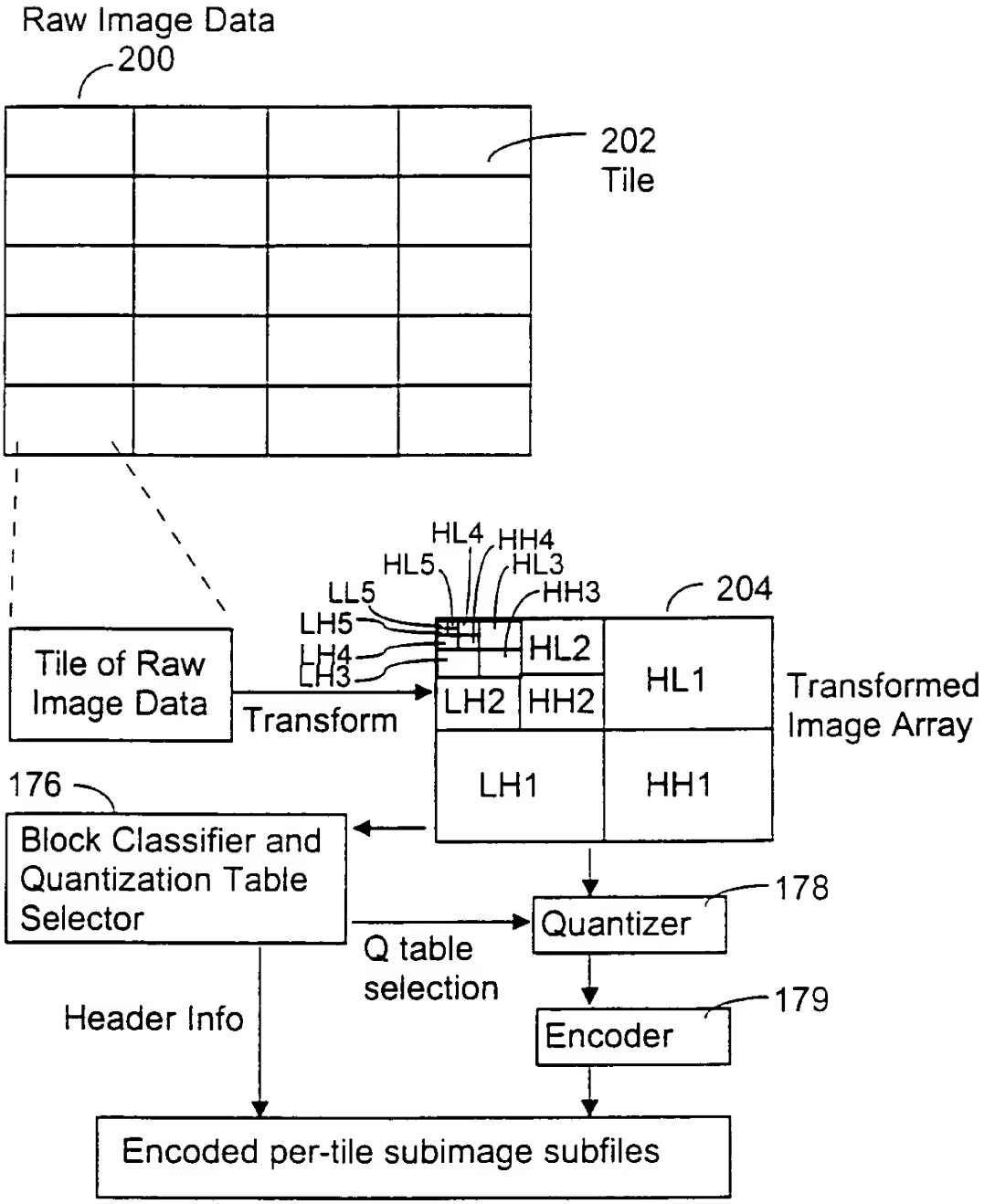

Raw Image Data
200
202
Tile
HL4
HH4
HL5
HL3
LL5
HH3
204
LH5
LH4
LH3
HL2
HL1
Tile of Raw Image Data
Transform
LH2
HH2
Transformed Image Array
LH1
HH1
176
Block Classifier and Quantization Table Selector
178
Quantizer
Q table selection
179
Header Info
Encoder
Encoded per-tile subimage subfiles

FIG. 6A

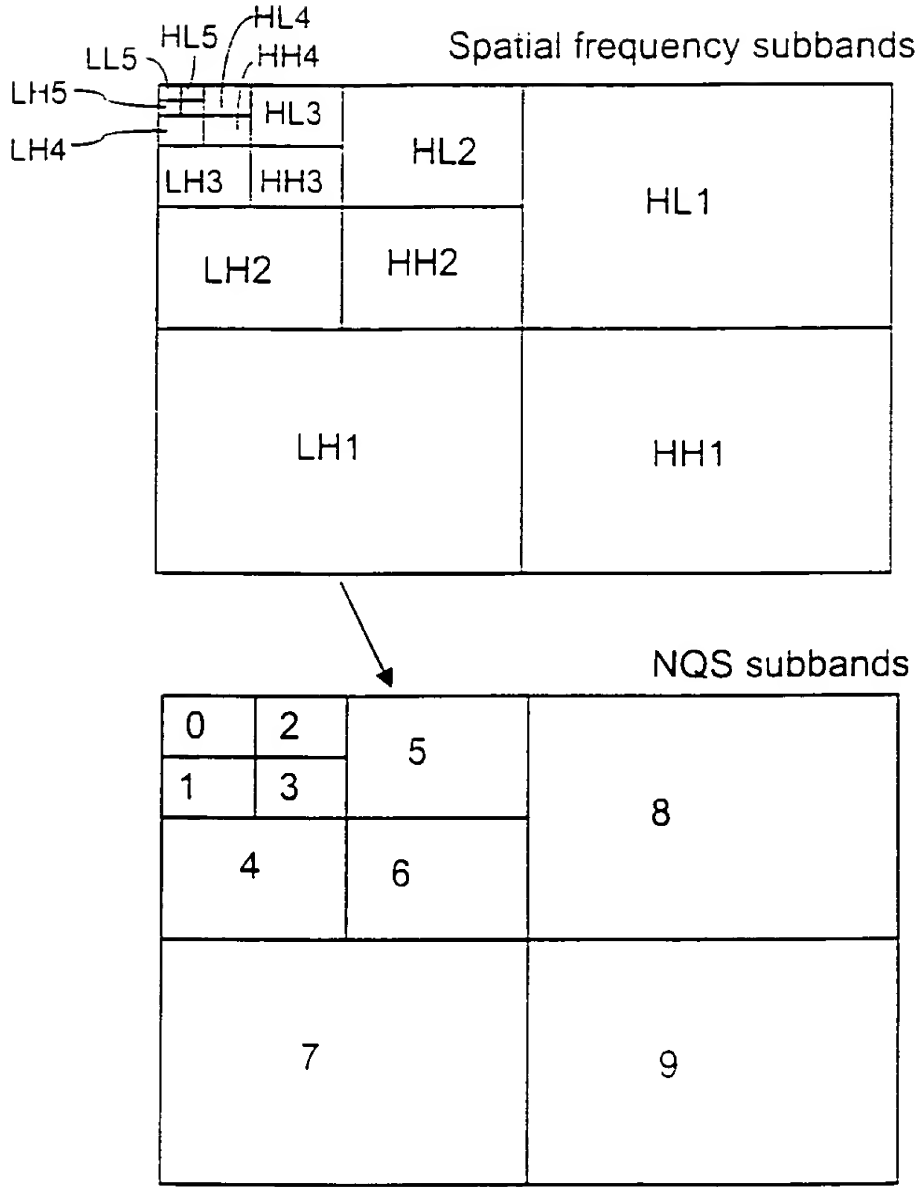


FIG. 6B

Figure 7

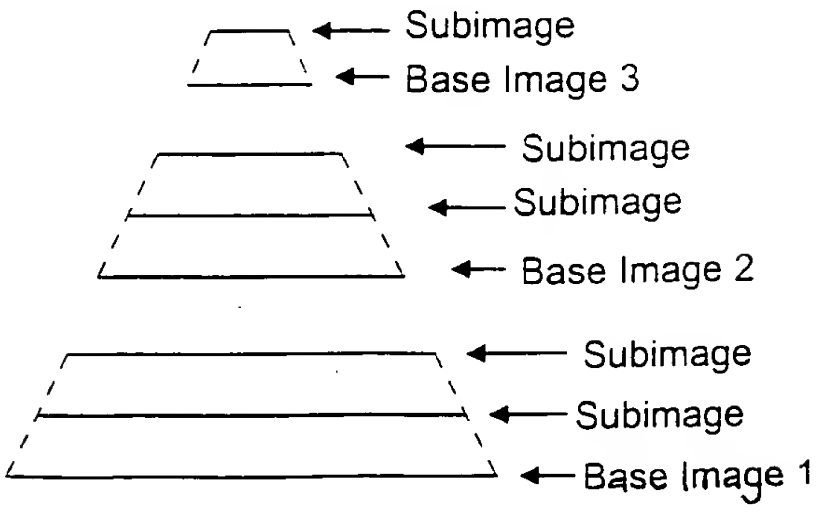


Figure 8A

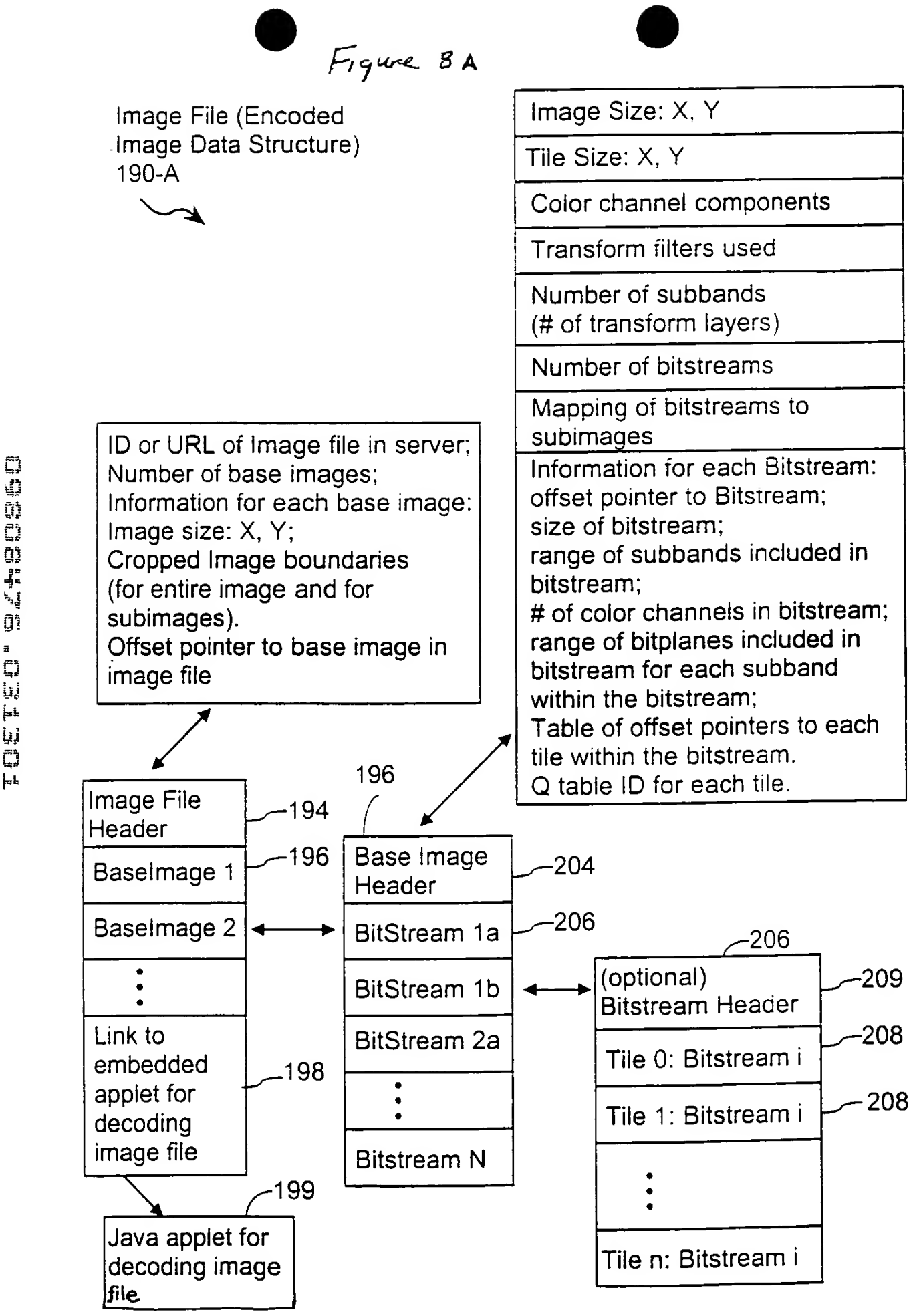


Image File (Encoded Image Data Structure) 190-A
Image Size: X, Y
Tile Size: X, Y
Color channel components
Transform filters used
Number of subbands (# of transform layers)
Number of bitstreams
Mapping of bitstreams to subimages
Information for each Bitstream: offset pointer to Bitstream; size of bitstream; range of subbands included in bitstream; # of color channels in bitstream; range of bitplanes included in bitstream for each subband within the bitstream; Table of offset pointers to each tile within the bitstream. Q table ID for each tile.
ID or URL of Image file in server; Number of base images; Information for each base image: Image size: X, Y; Cropped Image boundaries (for entire image and for subimages). Offset pointer to base image in image file
196
Image File Header
194
BaseImage 1
196
BaseImage 2
Link to embedded applet for decoding image file
198
Base Image Header
204
BitStream 1a
206
BitStream 1b
BitStream 2a
Bitstream N
206
(optional) Bitstream Header
209
Tile 0: Bitstream i
208
Tile 1: Bitstream i
208
Tile n: Bitstream i
199
Java applet for decoding image file

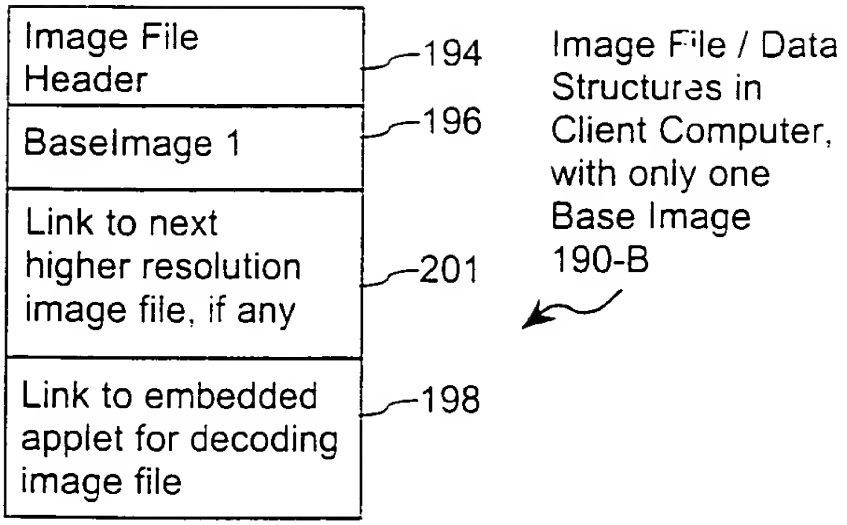


Image File Header
194
BaseImage 1
196
Link to next higher resolution image file, if any
201
Link to embedded applet for decoding image file
198
Image File / Data Structures in Client Computer, with only one Base Image 190-B

FIG. 8B

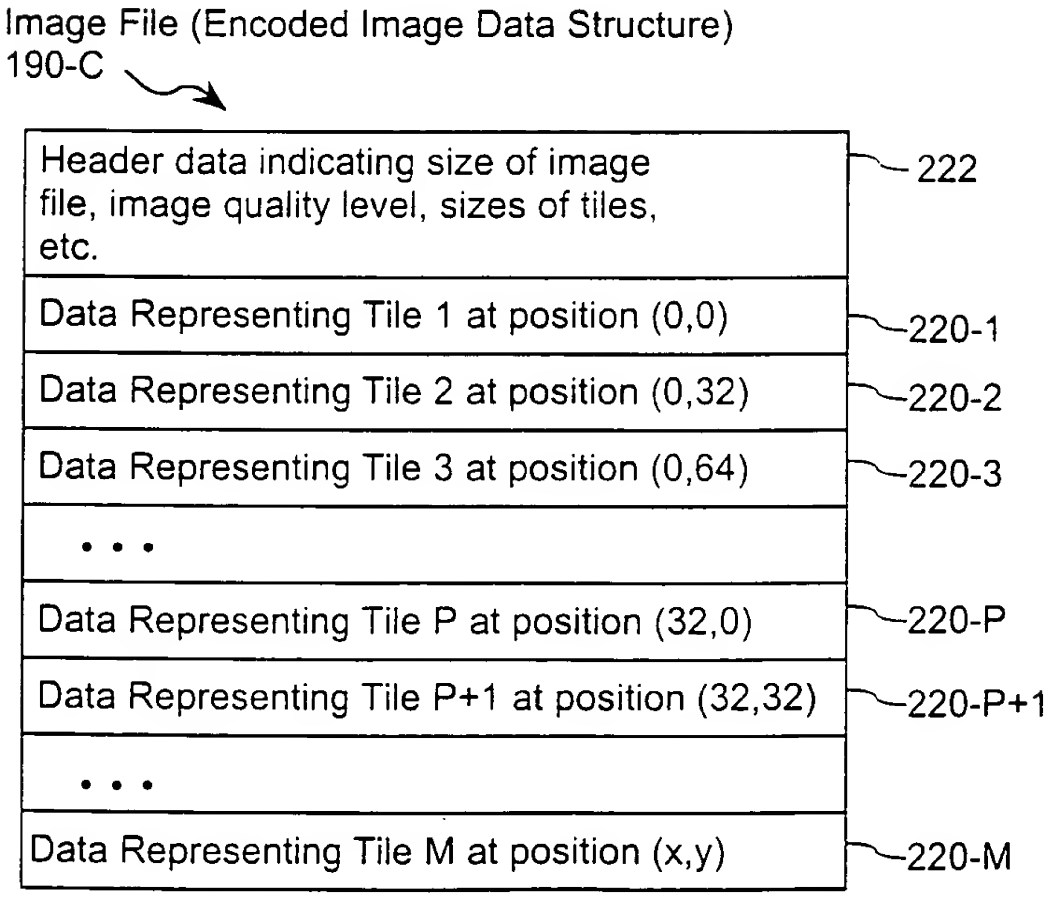


Image File (Encoded Image Data Structure) 190-C
Header data indicating size of image file, image quality level, sizes of tiles, etc.
222
Data Representing Tile 1 at position (0,0)
220-1
Data Representing Tile 2 at position (0,32)
220-2
Data Representing Tile 3 at position (0,64)
220-3
• • •
Data Representing Tile P at position (32,0)
220-P
Data Representing Tile P+1 at position (32,32)
220-P+1
• • •
Data Representing Tile M at position (x,y)
220-M

FIG. 8C

Data Representing One Tile t
220

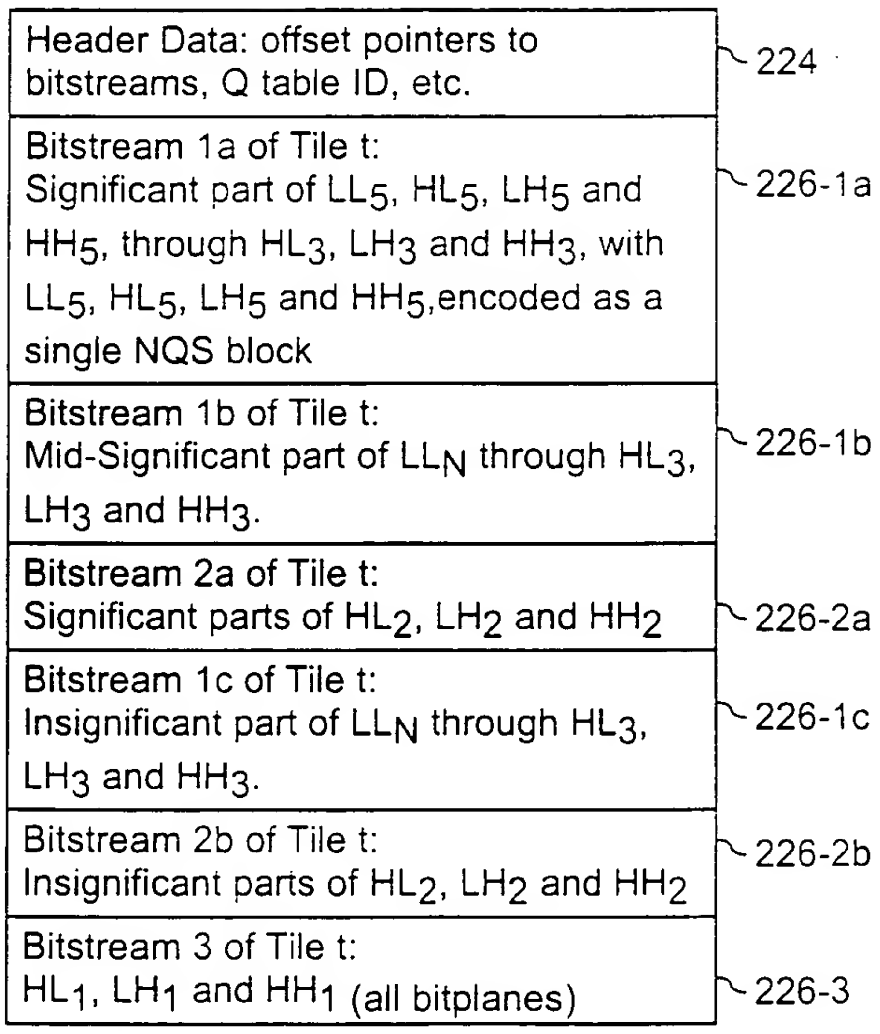


FIG. 8D

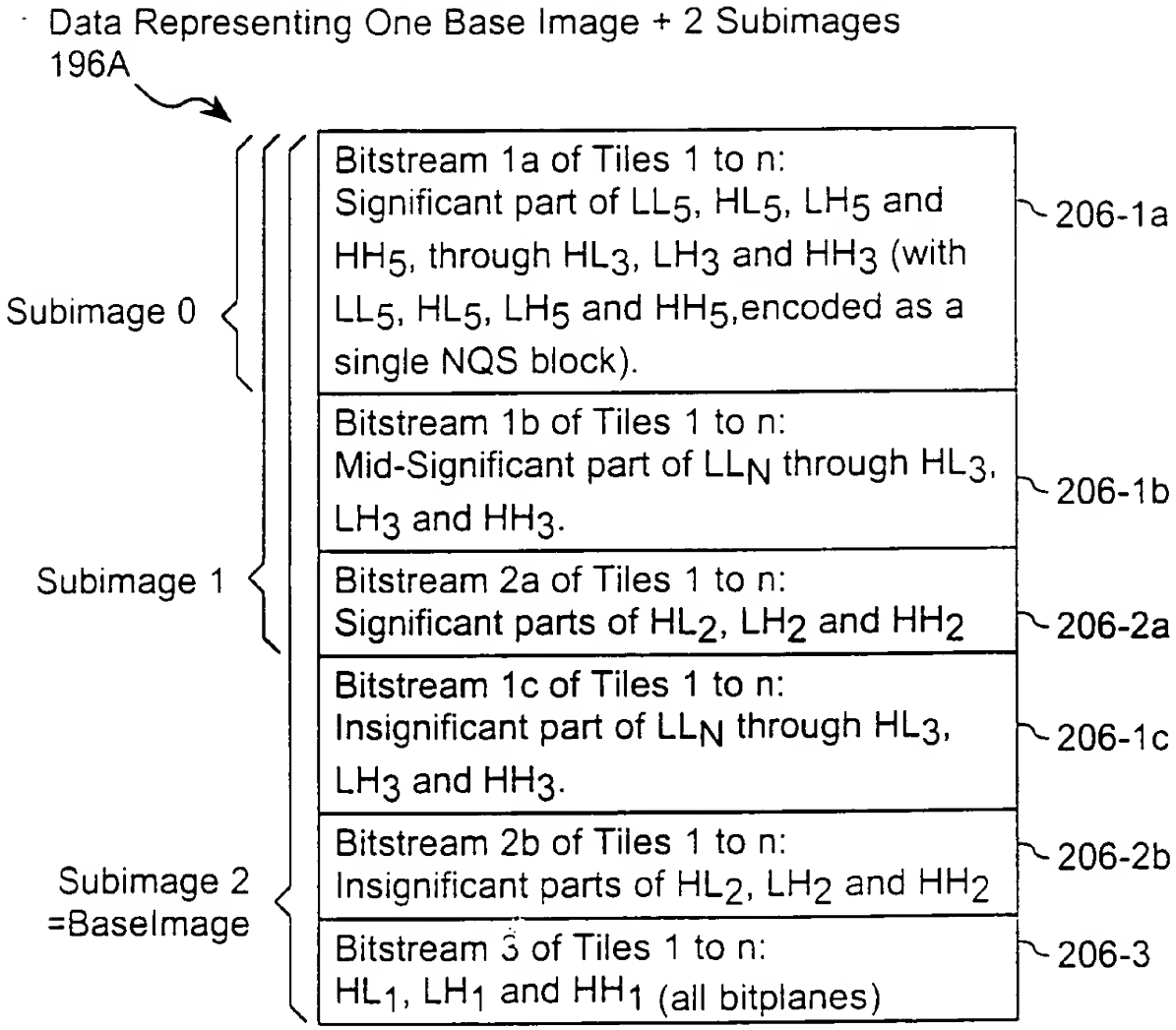

Data Representing One Base Image + 2 Subimages
196A
Subimage 0
Subimage 1
Subimage 2
=BaseImage
Bitstream 1a of Tiles 1 to n:
Significant part of LL5, HL5, LH5 and HH5, through HL3, LH3 and HH3 (with LL5, HL5, LH5 and HH5,encoded as a single NQS block).
206-1a
Bitstream 1b of Tiles 1 to n:
Mid-Significant part of LLN through HL3, LH3 and HH3.
206-1b
Bitstream 2a of Tiles 1 to n:
Significant parts of HL2, LH2 and HH2
206-2a
Bitstream 1c of Tiles 1 to n:
Insignificant part of LLN through HL3, LH3 and HH3.
206-1c
Bitstream 2b of Tiles 1 to n:
Insignificant parts of HL2, LH2 and HH2
206-2b
Bitstream 3 of Tiles 1 to n:
HL1, LH1 and HH1 (all bitplanes)
206-3

FIG. 8E

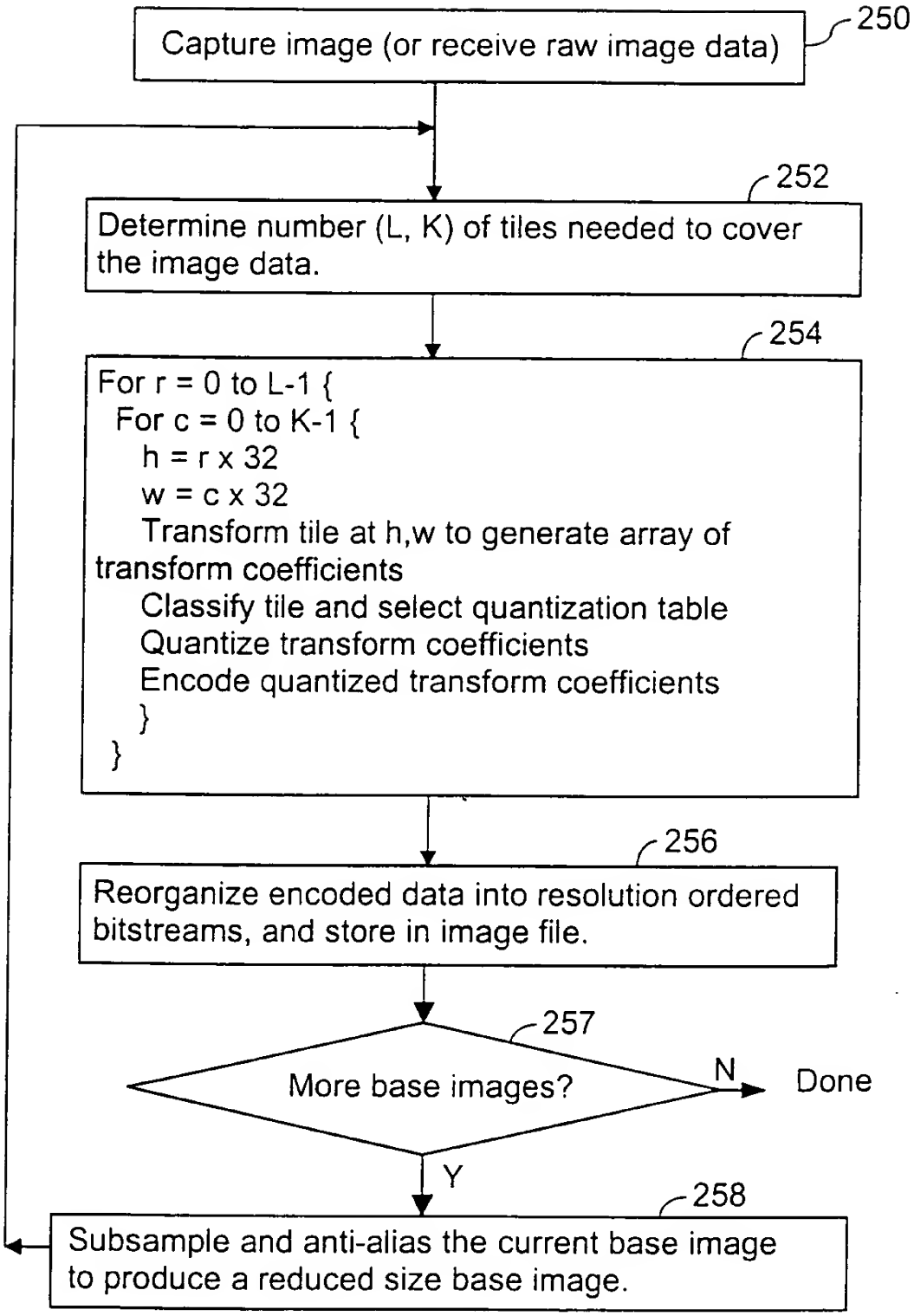

Capture image (or receive raw image data)
250
252
Determine number (L, K) of tiles needed to cover the image data.
254
For r = 0 to L-1 {
For c = 0 to K-1 {
h = r x 32
w = c x 32
Transform tile at h,w to generate array of transform coefficients
Classify tile and select quantization table
Quantize transform coefficients
Encode quantized transform coefficients
}
}
256
Reorganize encoded data into resolution ordered bitstreams, and store in image file.
257
More base images?
N
Done
Y
258
Subsample and anti-alias the current base image to produce a reduced size base image.

FIG. 9

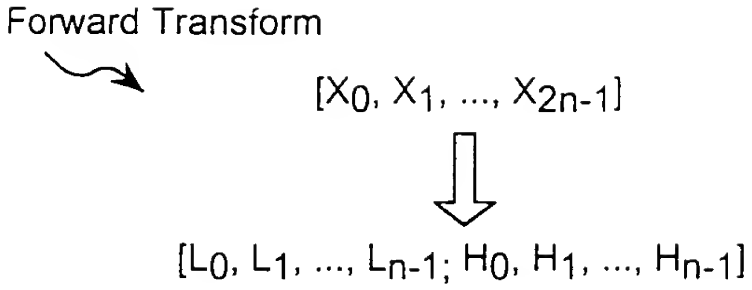


FIG. 10A

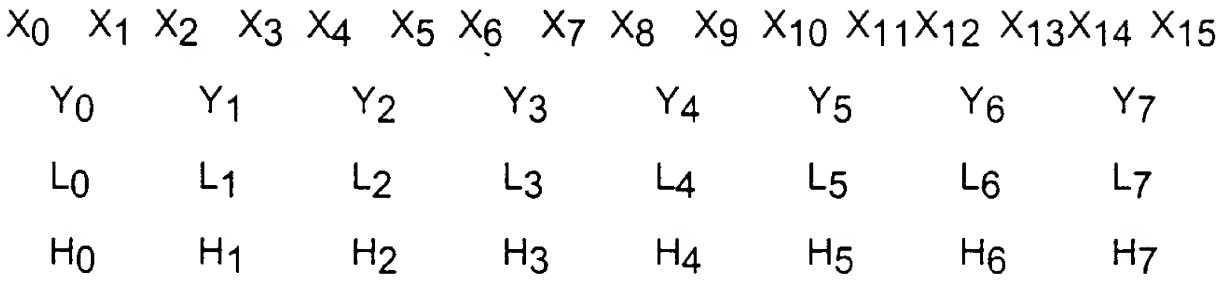


FIG. 10B

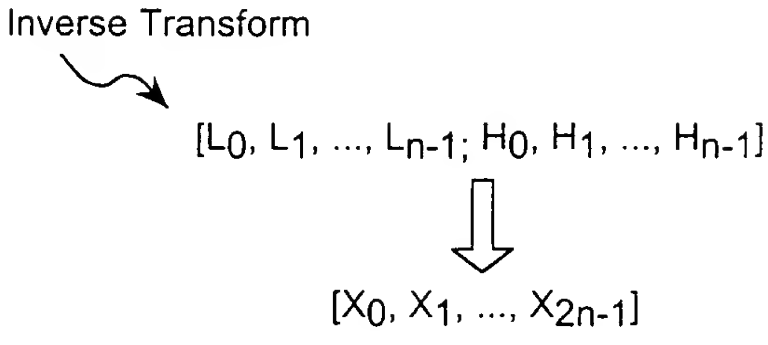


FIG. 10C

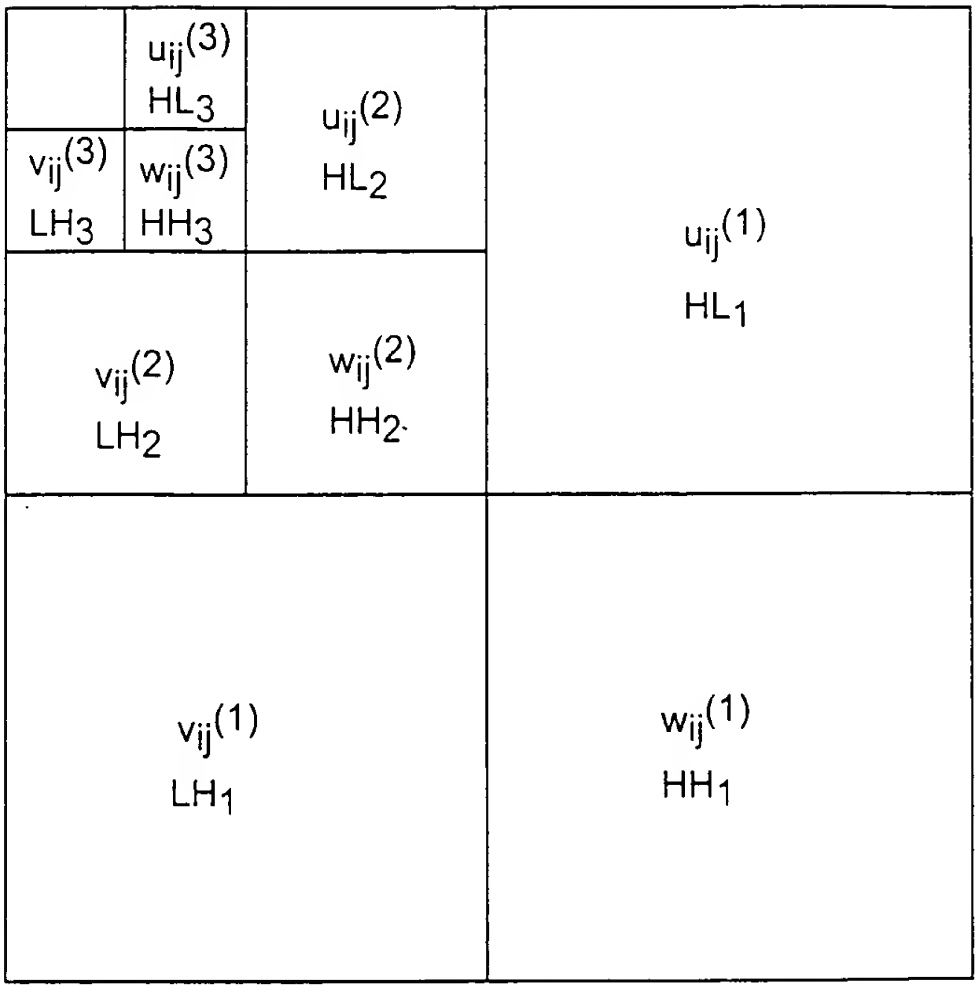

$u_{ij}^{(3)}$ HL_3
$u_{ij}^{(2)}$ HL_2
$u_{ij}^{(1)}$ HL_1
$v_{ij}^{(3)}$ LH_3
$w_{ij}^{(3)}$ HH_3
$v_{ij}^{(2)}$ LH_2
$w_{ij}^{(2)}$ HH_2
$v_{ij}^{(1)}$ LH_1
$w_{ij}^{(1)}$ HH_1

FIG. 11

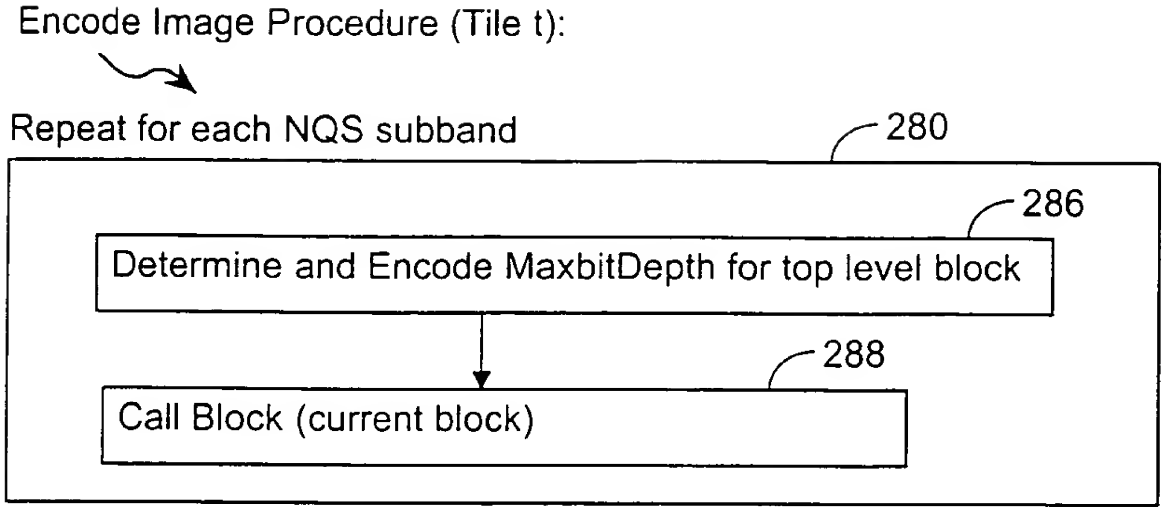
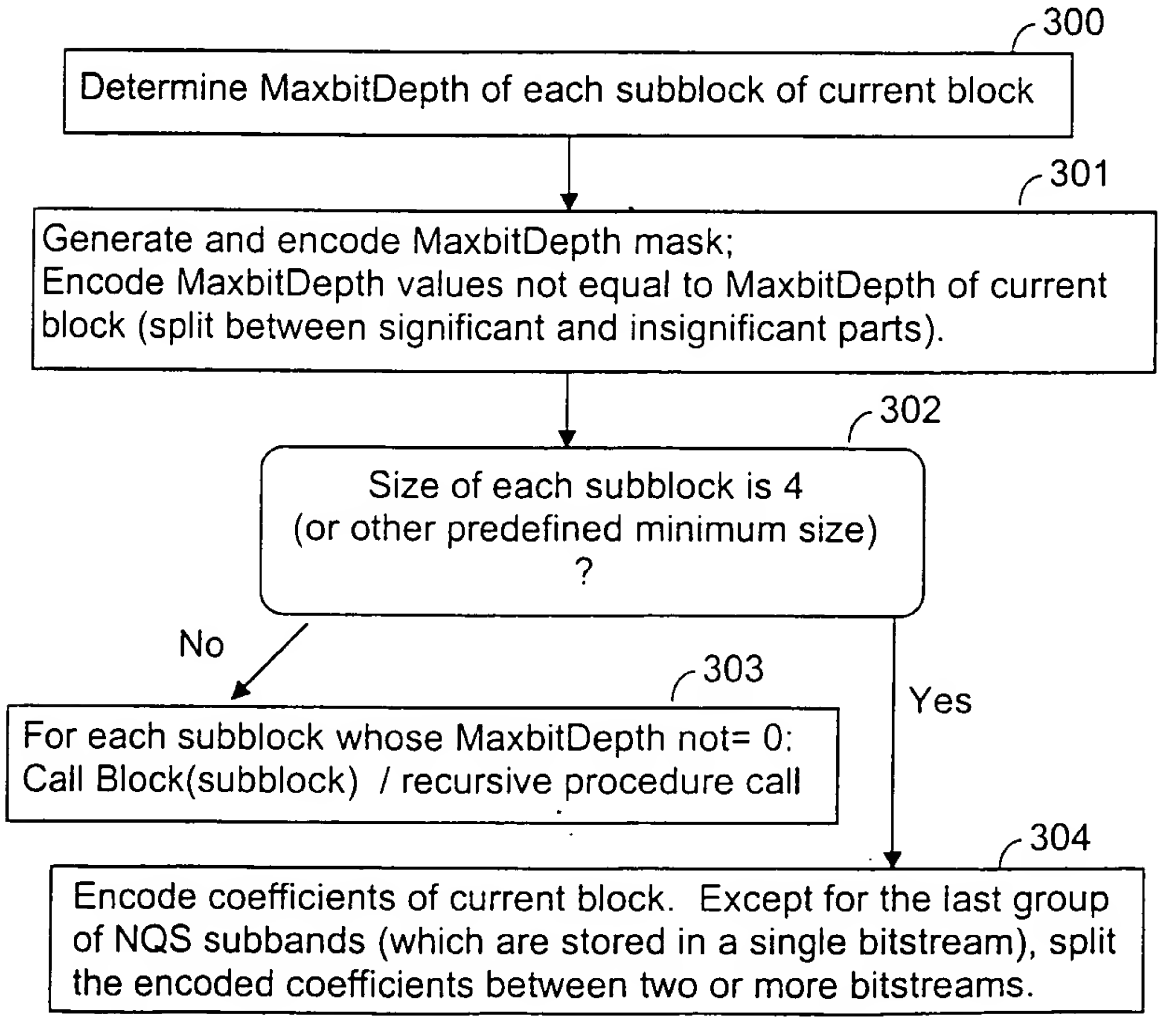


FIG. 13A

Block Procedure:

Fig 13B



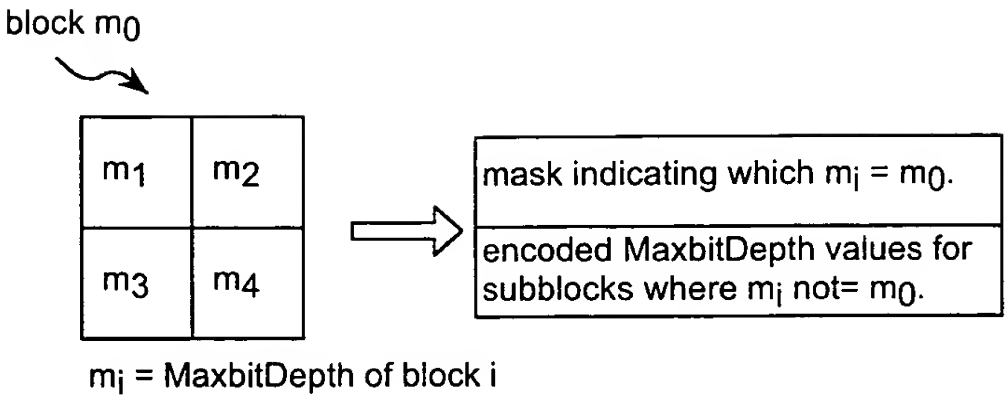


FIG. 14 A

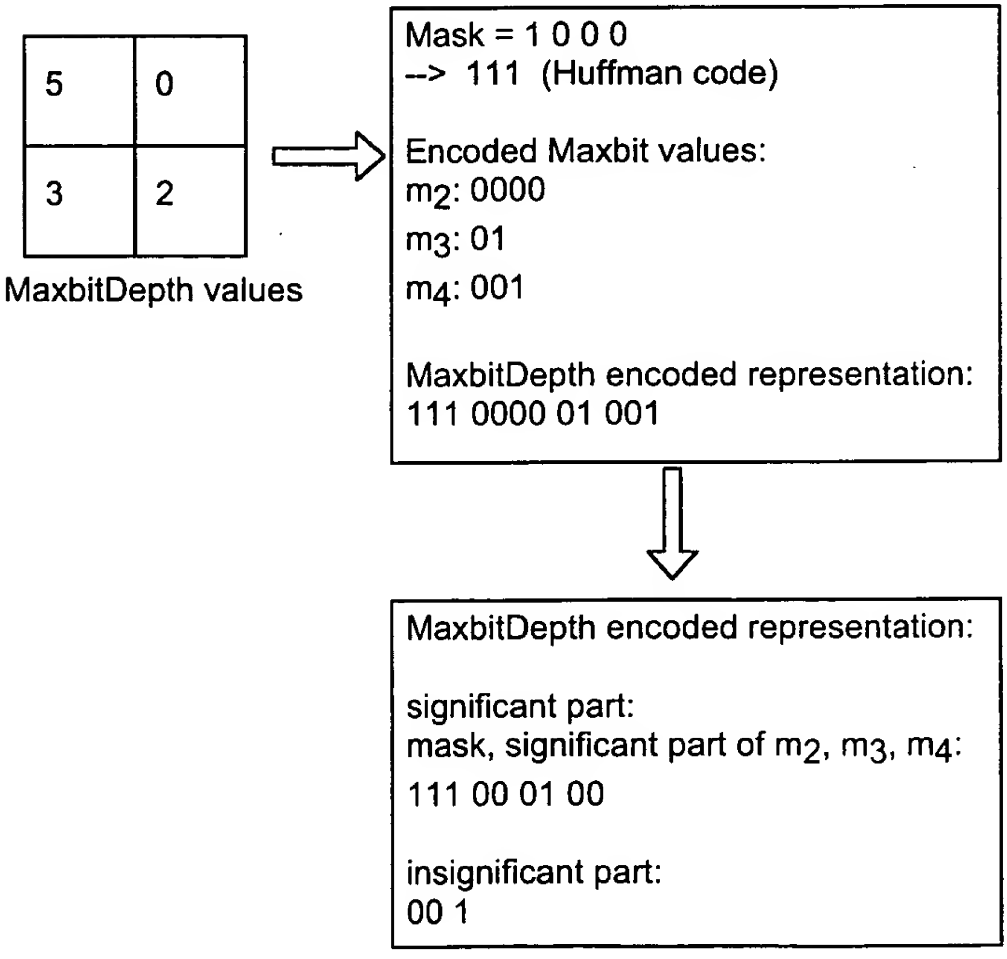


FIG. 14 B

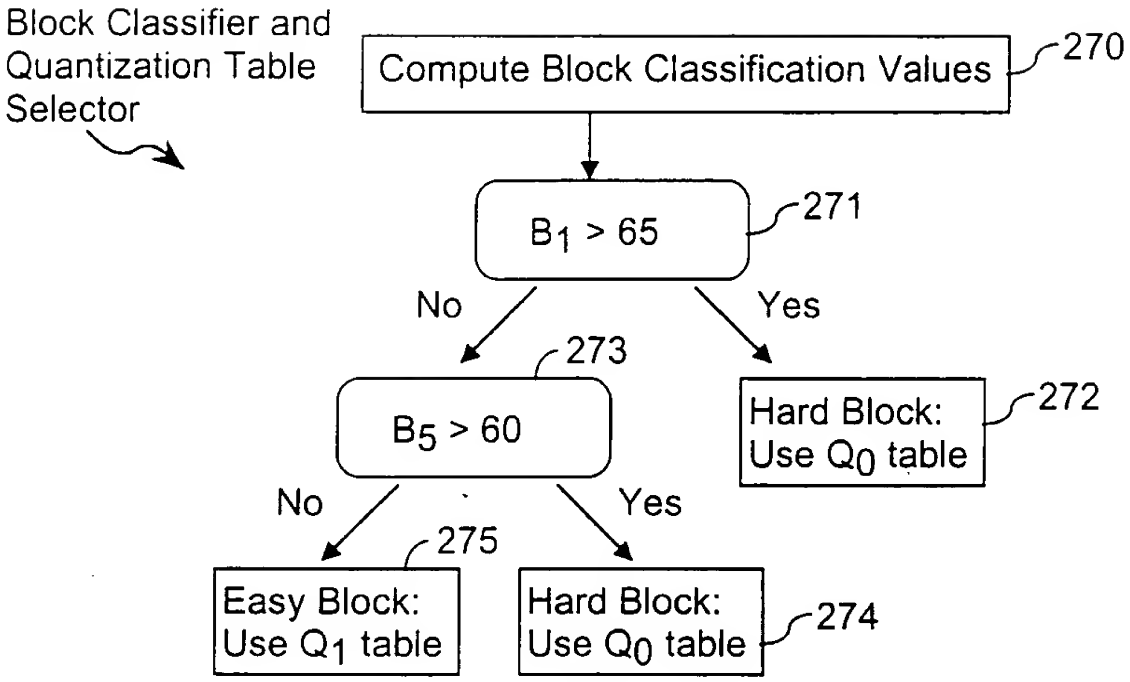


FIG. 12

Example

Encoding of Coefficients: 31, 0, -5, -2

significant part (threshold=3 bits):
POS 1, NEG

insignificant part:
111, 01, NEG 0

(MSB of each coefficient is known from MaxbitDepth values)

FIG. 14C

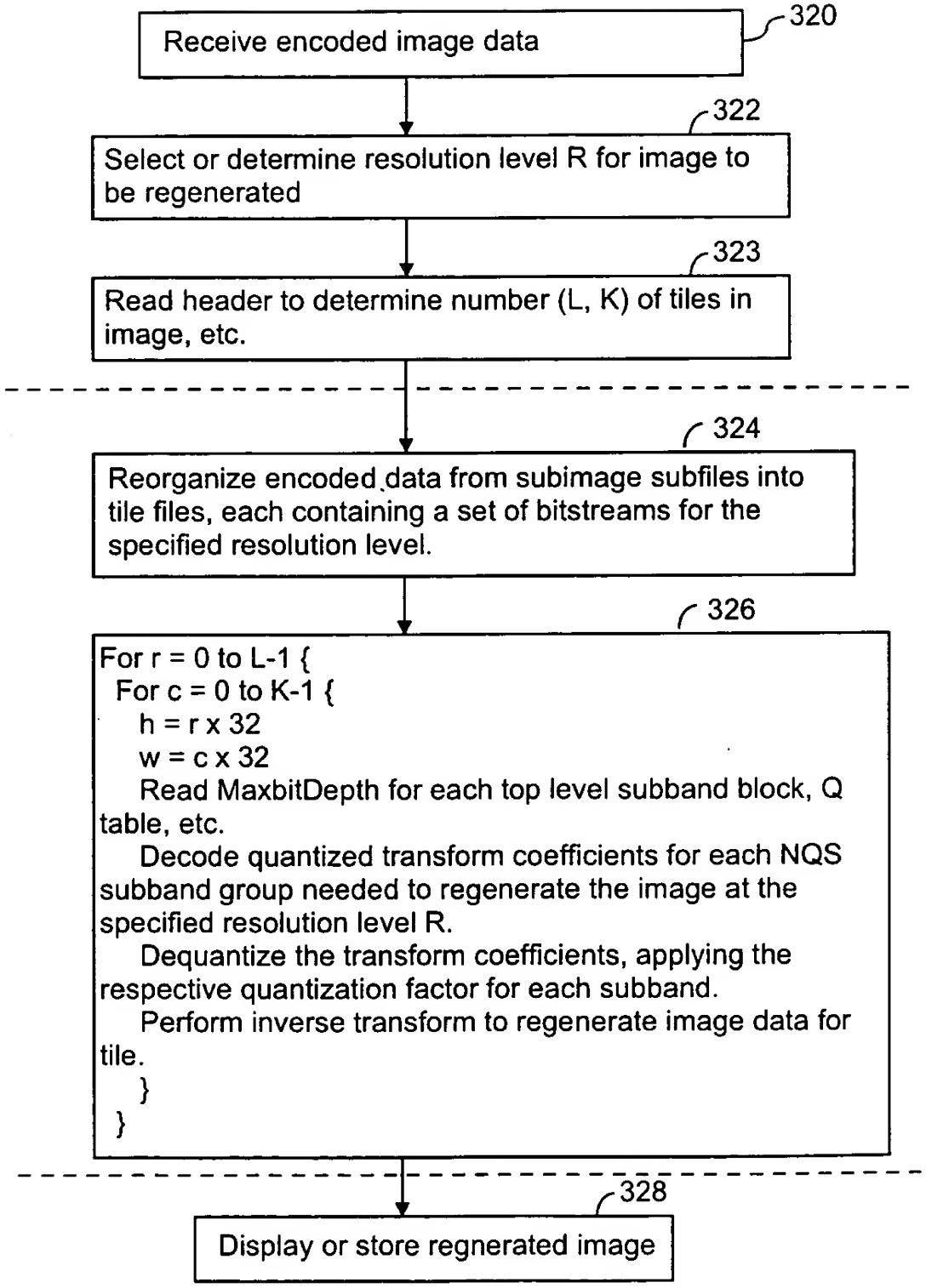

Receive encoded image data
320
Select or determine resolution level R for image to be regenerated
322
Read header to determine number (L, K) of tiles in image, etc.
323
Reorganize encoded data from subimage subfiles into tile files, each containing a set of bitstreams for the specified resolution level.
324
For r = 0 to L-1 {
For c = 0 to K-1 {
h = r x 32
w = c x 32
Read MaxbitDepth for each top level subband block, Q table, etc.
Decode quantized transform coefficients for each NQS subband group needed to regenerate the image at the specified resolution level R.
Dequantize the transform coefficients, applying the respective quantization factor for each subband.
Perform inverse transform to regenerate image data for tile.
}
}
326
Display or store regnerated image
328

FIG. 15

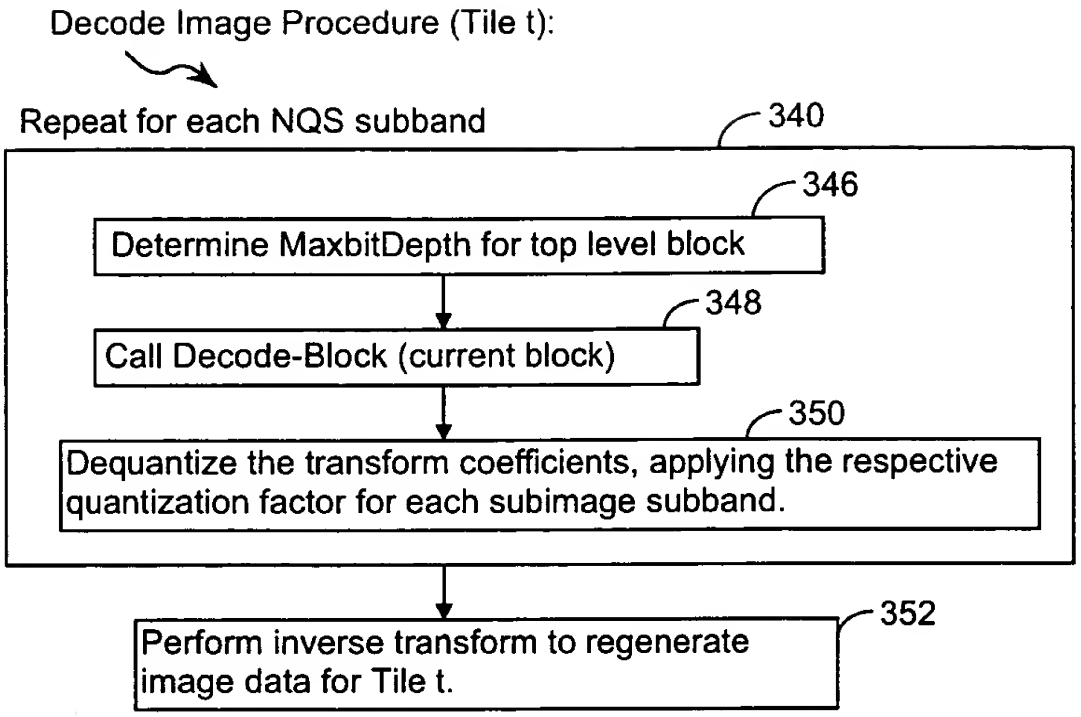


FIG. 16A

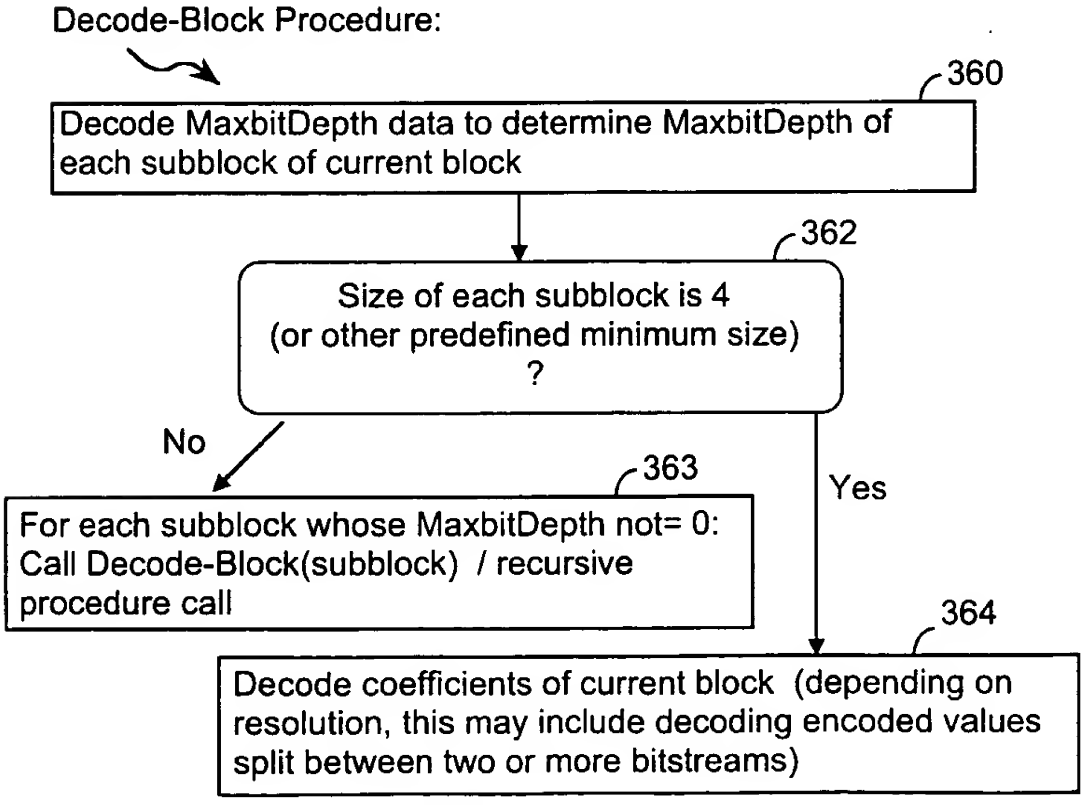


FIG. 16B

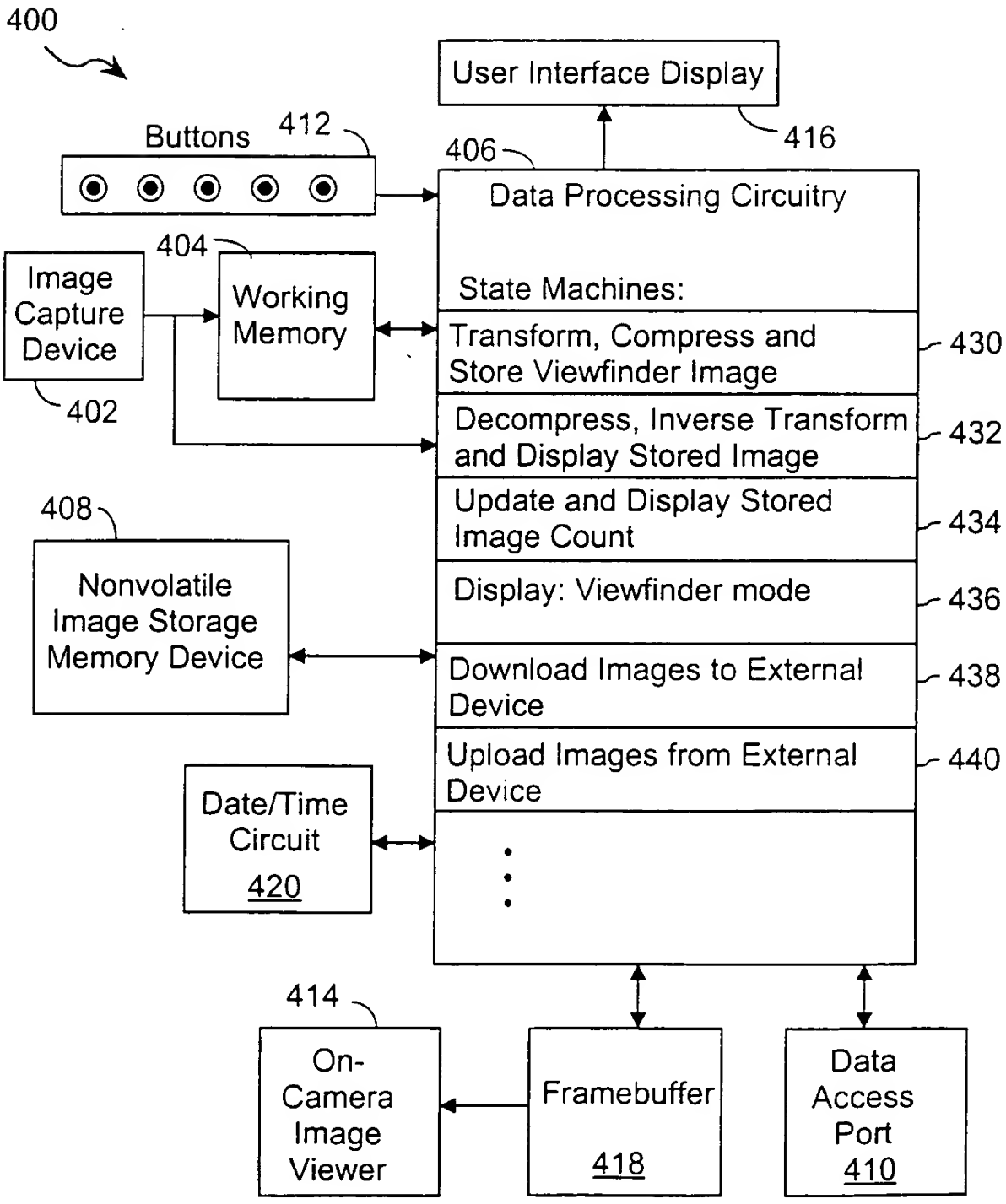

400
User Interface Display
416
Buttons
412
406
Data Processing Circuitry
State Machines:
404
Image Capture Device
Working Memory
402
Transform, Compress and Store Viewfinder Image
430
Decompress, Inverse Transform and Display Stored Image
432
Update and Display Stored Image Count
434
408
Nonvolatile Image Storage Memory Device
Display: Viewfinder mode
436
Download Images to External Device
438
Upload Images from External Device
440
Date/Time Circuit
420
414
On-Camera Image Viewer
Framebuffer
418
Data Access Port
410

FIG. 17

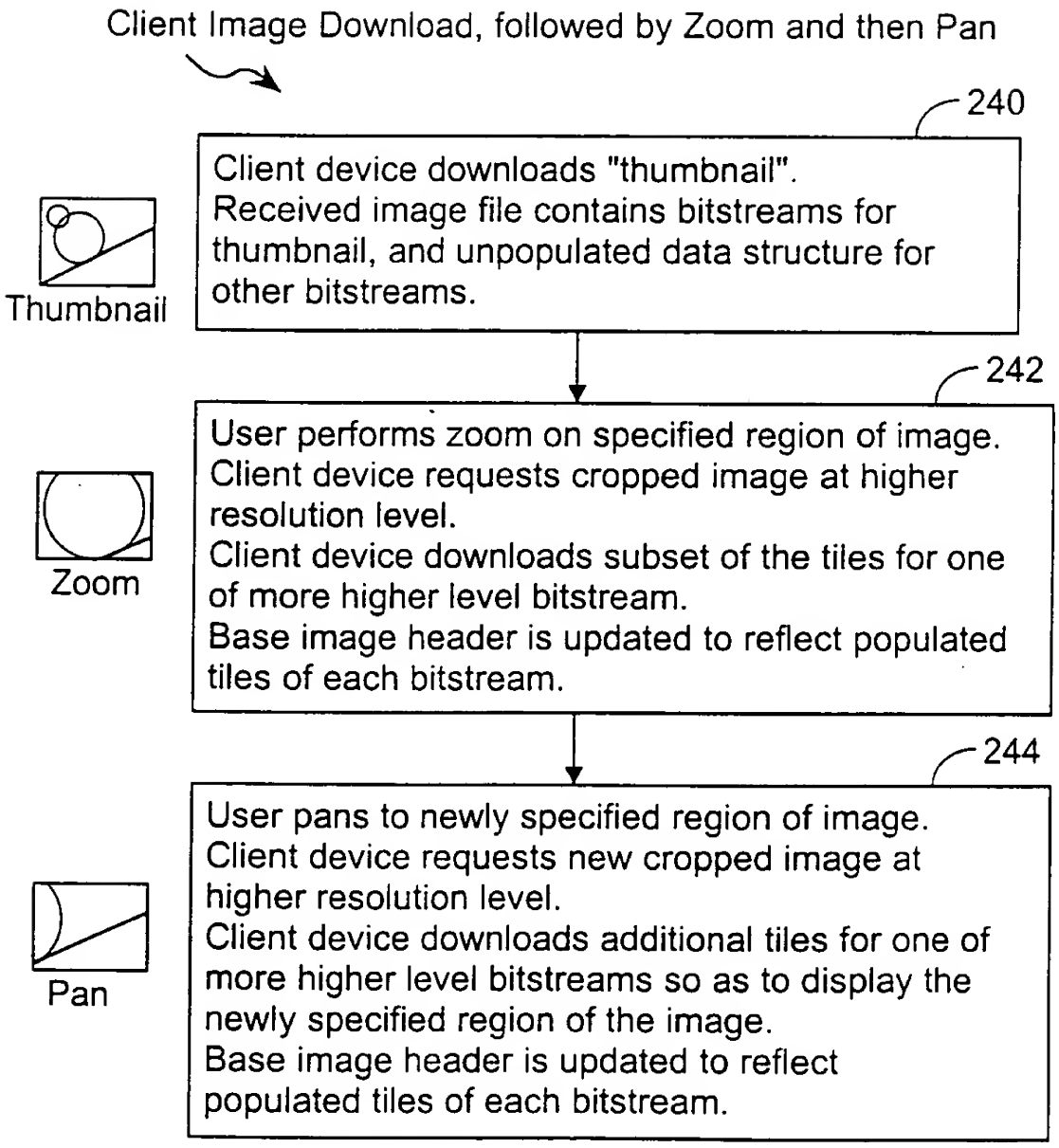

Client Image Download, followed by Zoom and then Pan
240
Client device downloads "thumbnail".
Received image file contains bitstreams for thumbnail, and unpopulated data structure for other bitstreams.
Thumbnail
242
User performs zoom on specified region of image.
Client device requests cropped image at higher resolution level.
Client device downloads subset of the tiles for one of more higher level bitstream.
Base image header is updated to reflect populated tiles of each bitstream.
Zoom
244
User pans to newly specified region of image.
Client device requests new cropped image at higher resolution level.
Client device downloads additional tiles for one of more higher level bitstreams so as to display the newly specified region of the image.
Base image header is updated to reflect populated tiles of each bitstream.
Pan

FIG. 18